Notes for Min-Cut and Max-Cut

ritchie huang

2019-09-20

Contents

1 Understand Karger's Min-Cut Algorithm

Let **G** be the graph with **n** nodes, **C** be the min-cut.

1.1 Pseudocode for Karger's Min-cut algorithm

Algorithm 1 Min-Cut

Input: graph **G(V, E)**
Output: min-cut **C**

```
while |V| > 2 do
    choose a uniform edge e ∈ E;
    contract(e);
return set of remaining edges C
```

1.2 Analysis of Karger's Algorithm

Theorem 1

$$Pr\left[\boldsymbol{C} \text{ is returned}\right] \geq \frac{2}{n(n-1)}$$

Proof 1 *In the (i+1) th contraction,* $i = 0, 1, ..., n-3$*. There are only* $(n-i)$ *nodes in the graph* $\boldsymbol{G}$*, so the* ***edges space*** *to contract on is* ***at least*** $\frac{(n-i)|\boldsymbol{C}|}{2}$*. that is, if we denote the size of edges space as* m*, we have* $m \geq \frac{(n-i)|\boldsymbol{C}|}{2}$*.*

Therefore, we obtain:

$$\begin{aligned} Pr\left[\text{sets of edges contracted in } (i+1) \text{ th step} \cap \boldsymbol{C} = \emptyset\right] &= 1 - \frac{|\boldsymbol{C}|}{m} \\ &\geq 1 - \frac{|\boldsymbol{C}|}{\frac{(n-i)|\boldsymbol{C}|}{2}} \\ &= 1 - \frac{2}{n-i} \end{aligned} \tag{1}$$

Combining all the $(n-2)$ *steps together:*

$$
\begin{aligned}
Pr\left[return\, \boldsymbol{C}\right] &= Pr\left[sets\ of\ edges\ contracted\ in\ (n\ \text{-}\ 2)\ steps \cap \boldsymbol{C} = \emptyset\right] \\
&\geq \prod_{i=0}^{n-3}(1 - \frac{2}{n-i}) \qquad (2) \\
&= \frac{2}{n(n-1)}
\end{aligned}
$$

If we run the algorithm independently for $\frac{n(n-1)}{2}\log n$ times.

$$
\begin{aligned}
Pr\left[\text{return } \mathbf{C} \textbf{ at least once}\right] &= 1 - Pr\left[\text{all runs fail to return } \mathbf{C}\right] \\
&\geq 1 - \left(1 - \frac{2}{n(n-1)}\right)^{\frac{n(n-1)}{2}\log n} \qquad (3) \\
&= 1 - \left(\frac{1}{e}\right)^{logn} \\
&= 1 - \frac{1}{n}
\end{aligned}
$$

According to the proof, we are expected to get the correct min-cut with probability **more than** $1 - \frac{1}{n}$.

2 Understand Karger-Stein's Algorithm

2.1 Analysis of Karger-Stein's Algorithm

The key mind to get a faster algorithm is to stop the old algorithm at k-th contraction in a single run, where k is to be specified. How can we understand this **early stop** ?

If we expand the equation 2:

$$
Pr\left[\text{return } \mathbf{C}\right] = (1 - \frac{2}{n})(1 - \frac{2}{n-1})(1 - \frac{2}{n-2}) \cdots (1 - \frac{2}{4})(1 - \frac{2}{3}) \qquad (4)
$$

From the right-hand side, we can find that with the increase of terms, the probability changes accordingly.

Observing this image1, it's easy to sense that the curve declines more and more slowly as the number of remaining nodes decreases. Thus, we can stop the contraction when the remaining nodes is $\mathbf{t}$, the procedure contraction can be adaptively changed.

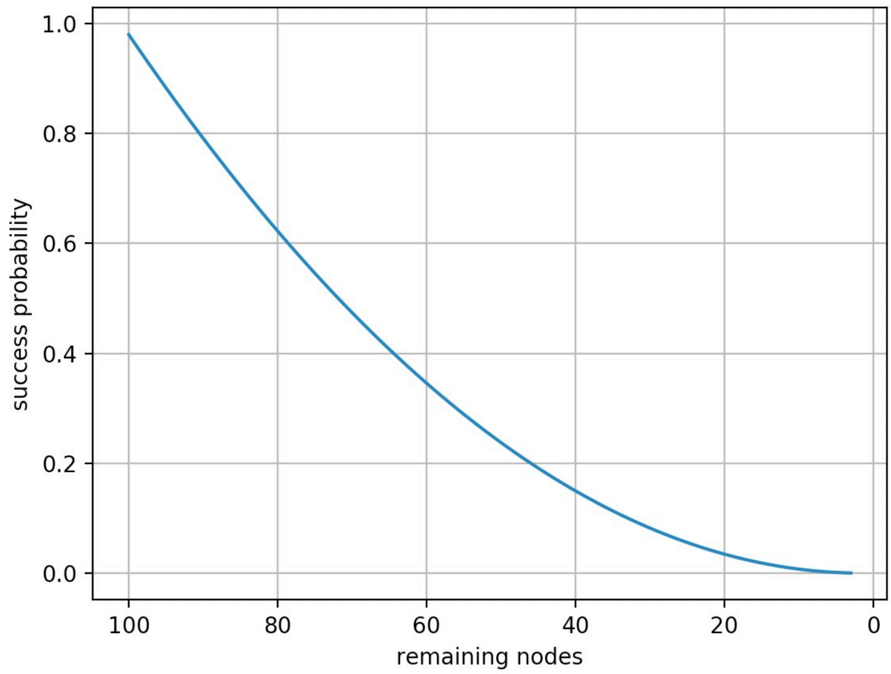

Figure 1: success probability vs. remaing nodes

This leaves us a question: What should **t** exactly be ?

In the original paper, Karger and Stein let $t = \lceil \frac{n}{\sqrt{2}} + 1 \rceil$.Because they hope to make equation 4 return a probability $\geq \frac{1}{2}$.Denoting the set of contracted edges util remaining **t** nodes as **A**.

$$\begin{aligned} Pr\left[\mathbf{A} \cap \mathbf{C} = \emptyset\right] &\geq \prod_{i=0}^{n-t-1} (1 - \frac{2}{n-i}) \\ &= \frac{t(t-1)}{n(n-1)} \\ &\geq \frac{(t-1)^2}{(n-1)^2} \\ &\geq \frac{1}{2} \end{aligned} \tag{5}$$

Solve the equation above, we get $t = \lceil \frac{n}{\sqrt{2}} + 1 \rceil$.

As we use the improved contraction algorithm above, we can use accordingly modify the old Min-Cut algorithm to Fast-Cut algorithm:

What's the probability that Fast-Cut Algorithm return the correct **C** ? For graph $\mathbf{G(V, E), |V| = n}$, we define the probability as $P(n)$ .As the 2

Algorithm 2 Improved Contraction Algorithm: Contract(**G, t**)

Input: graph **G**, contraction stop threshold **t**
Output: sets of remaining edges

```
while |V| > t do
    choose a uniform e ∈ E;
    contract(e);
return remaining edges in G
```

Algorithm 3 Karger-Stein's Algorithm: Fast-Cut(**G**)

Input: graph **G**, contraction stop threshold **t**
Output: number of remaining edges $|\hat{C}|$

```
1: set t = ⌈n/√2 + 1⌉
2: if |V| <= 6 then
3:     perform min-cut algorithm by brute-force.
4: else
5:     G1 = Contract(G, t);
6:     G2 = Contract(G, t);
7: return Min( Fast-Cut(G1), Fast-Cut(G2) )
```

contractions are independent, we obtain:

$$\begin{aligned} P(n) &= 1 - (1 - \frac{1}{2}P(\lceil \frac{n}{\sqrt{2}} + 1 \rceil))^2 \\ &\geq 1 - \left(1 - P(\lceil \frac{n}{\sqrt{2}} + 1 \rceil) + \frac{1}{4}P(\lceil \frac{n}{\sqrt{2}} + 1 \rceil)^2\right) \\ &= P(\lceil \frac{n}{\sqrt{2}} + 1 \rceil) - \frac{1}{4}P(\lceil \frac{n}{\sqrt{2}} + 1 \rceil)^2 \end{aligned} \tag{6}$$

By induction:

$$P(\lceil \frac{n}{\sqrt{2}} + 1 \rceil) = \Omega(\frac{1}{\log n})$$

Fast-Cut algorithm is ensured to return the correct min-cut **C** with probability **at least** $1 - \frac{1}{n}$:

$$\begin{aligned} 1 - (1 - P(\lceil \frac{n}{\sqrt{2}} + 1 \rceil))^{\log^2 n} &\geq 1 - (1 - \frac{1}{\log n})^{\log^2 n}, \quad n \to \infty \\ &= 1 - \left(\frac{1}{e}\right)^{logn} \\ &= 1 - \frac{1}{n} \end{aligned} \tag{7}$$

2.2 Time complexity analysis of Fast-Cut Algorithm

$$T(n) = 2T(\lceil \frac{n}{\sqrt{2}} + 1 \rceil) + O(n^2) \tag{8}$$

Applying **Master Theorem**, we can obtain that :

$$T(n) = O(n^2 \log n)$$